Instrucciones

Para el back-end se pide construir una API REST usando Java Spring Boot o Node Js que exponga la información al Pokedex.  
  
Esta API debe consumir el servicio externo PokeApi [https://pokeapi.co/](https://www.google.com/url?q=https://pokeapi.co/&sa=D&source=editors&ust=1644955994831305&usg=AOvVaw3GyeLx_Nz6CYsBSaX6WwJ_) para obtener la información. La documentación la puedes encontrar en [https://pokeapi.co/docs/v2](https://www.google.com/url?q=https://pokeapi.co/docs/v2&sa=D&source=editors&ust=1644955994831459&usg=AOvVaw2C72wyyZUv8Dpydv6z9Lm3).   
  
Se deben exponer los endpoints necesarios para poder montar un frontend (no es parte del entregable) que es una página donde se listan los pokemons inicialmente mostrando la siguiente información (información básica):  
- Foto (url)  
- Tipo (type)  
- Peso (weight)  
- Listado de Habilidades (ability)  
  
Cuando el usuario haga click en alguno de los pokemons, se muestra una pantalla con más detalles del pokemon seleccionado:  
- Información Básica (devuelta en la lista de pokemons)  
- Descripción (en español)  
- Lista de Movimientos (campo name en moves)  
  
- Debes incluir Swagger en tu API para poder probar los endpoints. Pero si deseas (opcional) puedes hacer un front-end que haga las llamadas mencionadas arriba, y puedes elegir el lenguaje de tu preferencia.   
  
La aplicación debe ser desplegada en AWS, Azure, GCP o Heroku, a elección. El código de la aplicación debe estar alojado en algún repositorio Git al que debes darnos acceso.  
  
Cualquier funcionalidad extra u optimización es bienvenida.   
  
Lo que evaluaremos con este desafío, es que seas capaz de:  
- Integrar una API externa.  
- Exponer un API usando Java Spring Boot.  
- Implementar Swagger o consumir API desde front-end  
- Desplegar en la nube.  
- Manejar errores.  
- Mantener un código ordenado y de calidad.  
  
Por favor subir el desafío a través de un repositorio público, ¡gracias!

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente